

Code for Pet.frm Pet.vbp

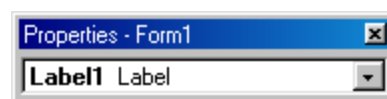


1. Double click on the Visual BASIC icon. If you were already in Visual BASIC, save your work then choose **File, New Project**.
2. Save your **Pet.frm** file from your gaggle digital locker to your desktop. Open it.
3. Click on the **View Form** button in the Project window.



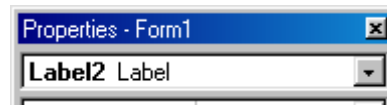
4. Your form should look similar to this.

5. Make sure that all of your labels except `lblTotal` have the **AutoSize** property set to **True**.
6. Click on the label that says **Pet Expense** at the top center of your form. Look at the top of the **Properties** window. It should say **Label1** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.

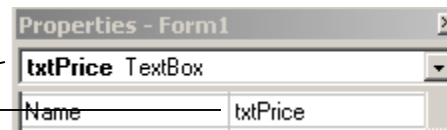
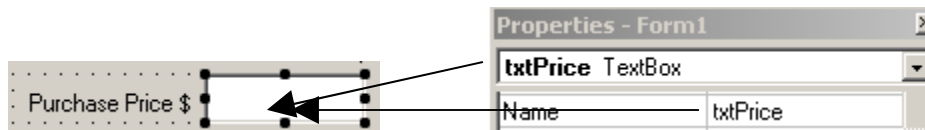


Code for Pet.frm Pet.vbp

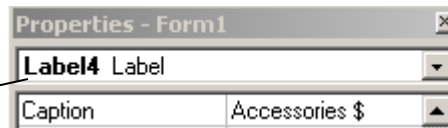
7. Select the label that says **Purchase Price \$**. In the **Properties** window, you see **Label2** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



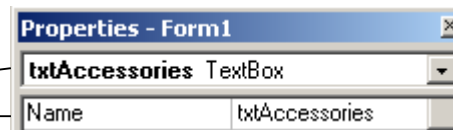
8. Click once on the white **TextBox** next to the words **Purchase Price \$**. In the **Properties** window, you should see **txtPrice** TextBox. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtPrice**.



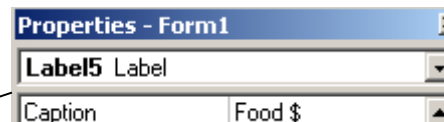
9. Select the label that says **Accessories**. In the **Properties** window, you see **Label4** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



10. Click once on the white **TextBox** next to the word **Accessories**. In the **Properties** window, you should see **txtAccessories** TextBox. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtAccessories**.



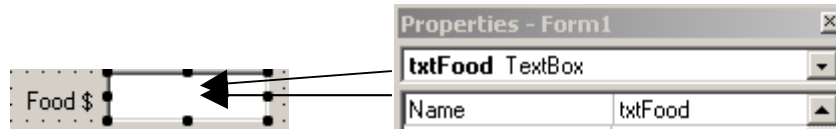
11. Select the label that says **Food \$**. In the **Properties** window, you see **Label5** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



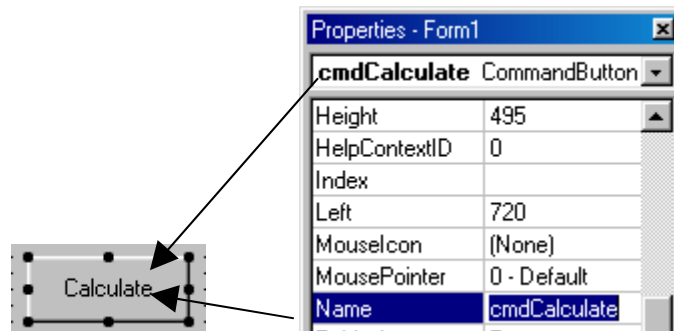
12. Click once on the white **TextBox** next to the word **Food \$**. In the **Properties** window, you should see **txtFood** TextBox. The word in **bold** is the name you gave the object. The word after that is


Code for Pet.frm Pet.vbp

the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtFood**.

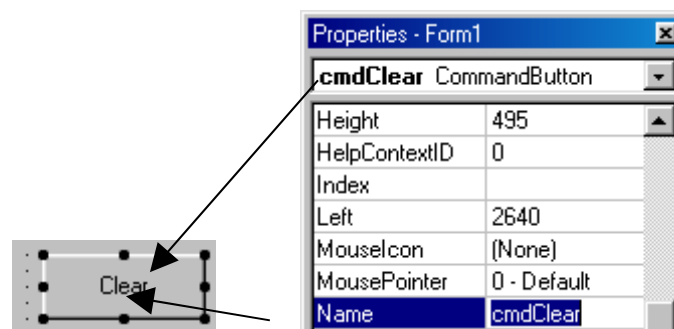


13. Click once on the **Calculate** CommandButton. Make sure the first **C** in **Calculate** is capital. In the **Properties** window, you should see **cmdCalculate** CommandButton. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **CommandButton** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **cmdCalculate**.



14. In the top toolbar click on the **Save** icon. 

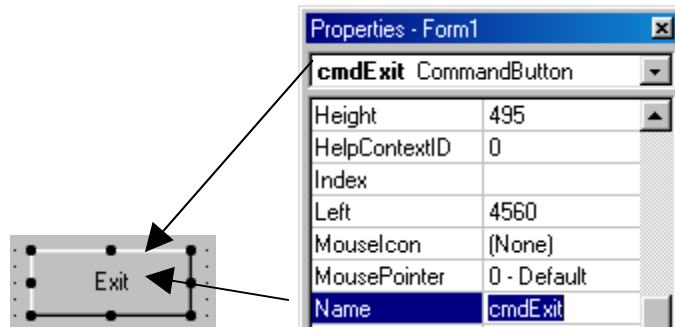
15. Click once on the **Clear** CommandButton. Make sure the first **C** in **Clear** is capital. In the **Properties** window, you should see **cmdClear** CommandButton. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **CommandButton** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **cmdClear**.



16. Click once on the **Exit** CommandButton. Make sure the first **E** in **Exit** is capital. In the **Properties** window, you should see **cmdExit** CommandButton. The word in **bold** is the name

Code for Pet.frm Pet.vbp

you gave the object. The word after that is the type of object it is. You need a specific name for this **CommandButton** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **cmdExit**.



17. Select the label that says **Total \$**. In the **Properties** window, you see **Label6 Label**. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



18. Select the gray box to the right of **Total \$**. This is a **Label** with no **Caption** currently. It looks like a box because you changed the **BorderStyle** Property to **Fixed Single**. In the **Properties** window, you see **lblTotal Label**. The word in **bold** is the name of the object. The word after that is the type of object it is.

You need a specific name for this **Label** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window.

Correct the **Name** property now, if it is not **lblTotal**.

(lowercase **L** as in **little**, lowercase **b** as in **boy**, lowercase **L** as in **label**, uppercase **A** as in **Apple**...)

This label is the place where the program will **OUTPUT** the results. When the program is run, the cursor will not land on labels because they are not for the user to change. They are only for the user to look at. If you can put the cursor in **lblTotal**, you probably made it a textbox by mistake. Always make sure the 3 letter prefix and the type of object correspond. If they do not, you probably have the wrong kind of object.

Make sure your properties window for the **lblTotal** label looks the same as the one pictured on the next page.

Code for Pet.frm Pet.vbp

Notice that its name is lblTotal and the type object is a label.

The box look is created by changing the BorderStyle to Fixed Single

Notice the caption has been deleted.

Properties - Form1	
lblTotal Label	
Alignment	0 - Left Justify
Appearance	1 - 3D
AutoSize	False
BackColor	&H8000000F&
BackStyle	1 - Opaque
BorderStyle	1 - Fixed Single
Caption	
DataField	
DataSource	
DragIcon	(None)
DragMode	0 - Manual
Enabled	True
Font	Comic Sans MS
ForeColor	&H80000012&
Height	495
Index	
Left	1920
LinkItem	
LinkMode	0 - None
LinkTimeout	50
LinkTopic	
MouseIcon	(None)
MousePointer	0 - Default
Name	lblTotal

19. In the top toolbar click on the Save icon.



Now you are ready to use the BASIC programming language code to make the command buttons do what the caption indicate (PROCESS the data). Since Visual BASIC is **event-driven**, the code is placed in **procedures** for the **event** that is used to trigger the action, which is often the **click event**.

Compare all the parts of the following pictures with what appears on your computer monitor to make sure you are in the correct **event procedure**. Look at the words in the windows pictured.

20. **Double click** on the Calculate CommandButton, so that the code window for the Object named **cmdCalculate** appears. The default Procedure for a command button is the Click event. The code in the Click Procedure is executed when the user **Clicks** on the Calculate command button.

Object: cmdCalculate Proc: Click

```
Private Sub cmdCalculate_Click()
|
End Sub
```

Calculate

Code for Pet.frm Pet.vbp

21. Put the cursor under `Private Sub cmdCalculate_Click()` and type the following code with the comments.

```

Dim Price As Single  'Declare variable for the price data input by the user in a textbox.
Dim Fee As Single    'Declare variable for the fee data input by the user in a textbox.
Dim Accessories As Single 'Declare variable for the accessories data input by the user.
Dim Food As Single   'Declare variable for the food data input by the user in a textbox.
Dim Total As Single  'Declare variable for total which is calculated by the program and the result
                     'is output to a label.

Price = Val(txtPrice) 'The text version of the price is being converted to a number and
                     'stored in the price variable in computer memory.

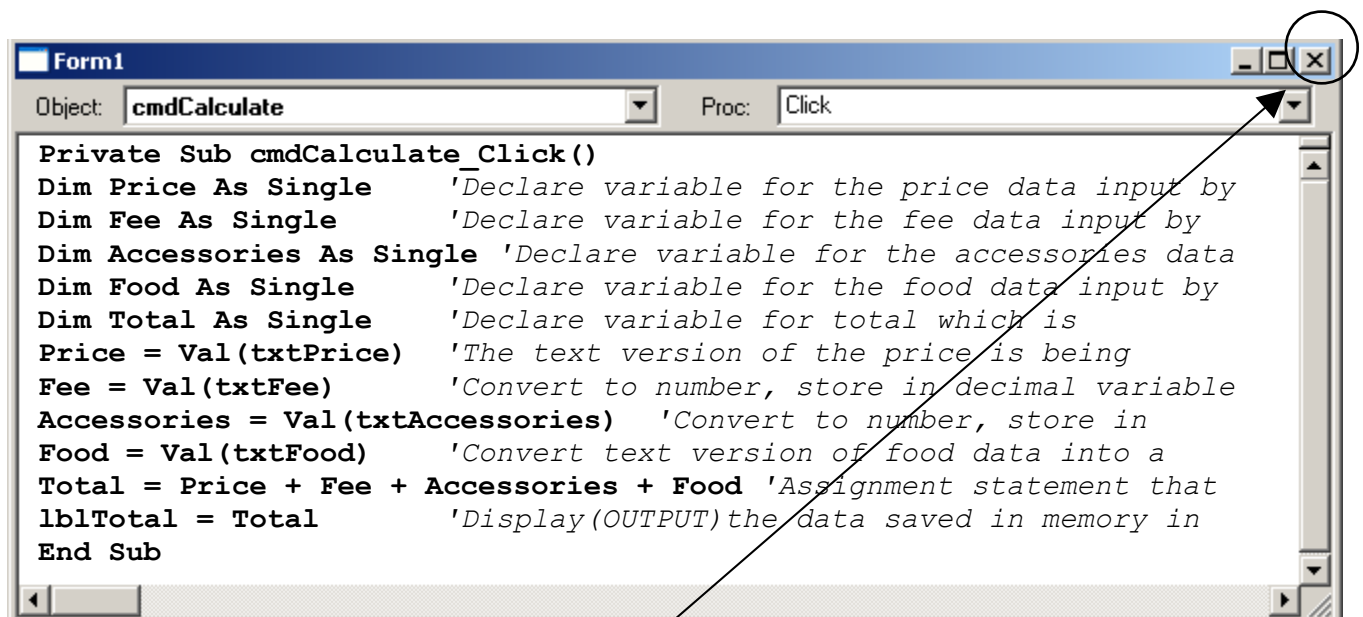
Fee = Val(txtFee)     'Convert to number, store in decimal variable in computer memory.

Accessories = Val(txtAccessories) 'Convert to number, store in memory.
Food = Val(txtFood)    'Convert text version of food data into a number, store in
memory.

Total = Price + Fee + Accessories + Food 'Assignment statement that calculates
the total and stores the result in the decimal variable, Total, in computer memory.
lblTotal = Total 'Display(OUTPUT) the data saved in memory in the label on the form.

```

It should look like this.



22. Close the code window using the .

In the top toolbar click on the Save icon.



Code for Pet.frm Pet.vbp

Question 1: What are these statements for?

```
Dim Price As Single
Dim Fee As Single
Dim Accessories As Single
Dim Food As Single
Dim Total As Single
```

Answer 1: The lines that start with `Dim` are variable declaration statements which are for telling the computer how much memory is necessary for storing the values of the variables.


`Single` is the variable type for decimal numbers that are not excessively large and do not have a lot of digits after the decimal point.

Question 2: What do these lines of code do?

```
Price = Val(txtPrice)
Fee = Val(txtFee)
Accessories = Val(txtAccessories)
Food = Val(txtFood)
```

Answer 2: The default variable type for text typed into a `TextBox` is the `String` type which holds characters. Numerical calculations cannot be done with `String` values; therefore, these strings of numbers will be converted to actual numbers by using the `Val` function. `Val` is short for *Value*. In other words the *text* is converted to a *numerical value*. The numerical values are then stored in the variables that have been designated as decimal numbers with the `Single` variable type.

```
Price = Val(txtPrice)
```

String (character) data converted to `Single` (decimal) data and stored in the appropriate variable.

Q3: What does this line of code do?

```
Total = Price + Fee + Accessories + Food
```

A3: Calculating a total is adding the numbers.

The variable that holds the answer must be on the left of the `=`.

The calculations on the right of the `=` are done first. Then the answer is stored in the memory location with the variable name which appears on the left of the `=`.

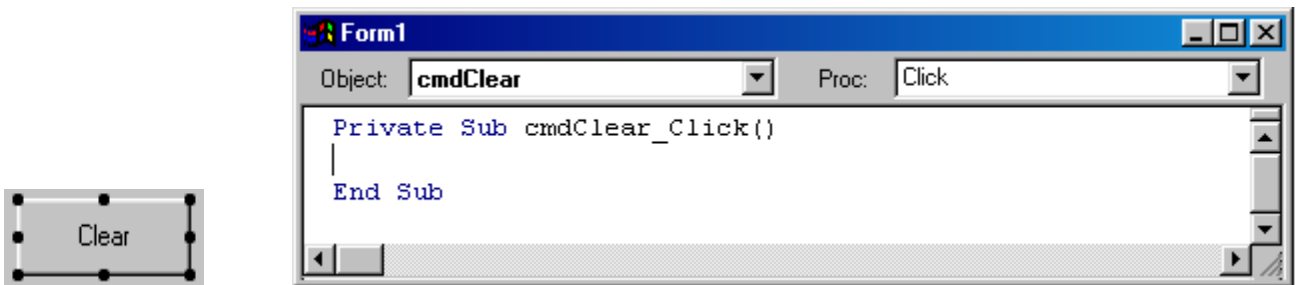
Code for Pet.frm Pet.vbp

Q4: What does this line of code do?

```
lblTotal = Total
```

A4: **lblTotal** is the variable name for the label that will output the display of the answer which is in the memory location called **Total**.

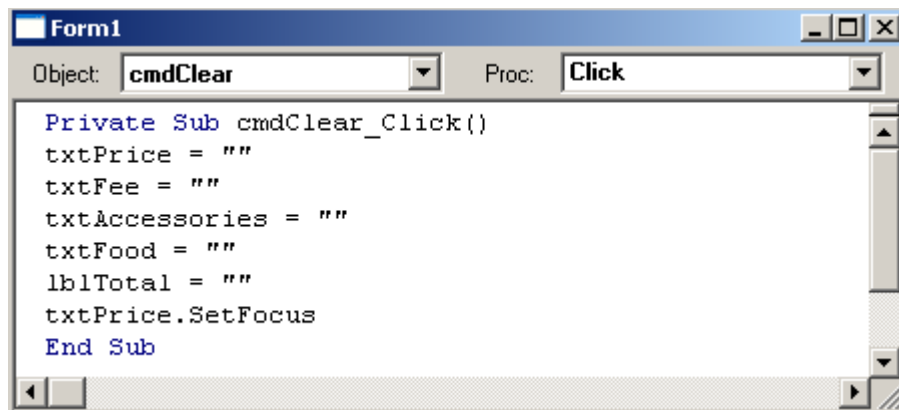
23. **Double click** on the Clear CommandButton, so that the code window for the Object named **cmdClear** appears. The default Procedure for a command button is the Click event. The code in the Click Procedure is executed when the user *Clicks* on the Clear command button.





24. Put the cursor under `Private Sub cmdClear_Click()` and type the following code:

```
txtPrice = ""
txtFee = ""
txtAccessories = ""
txtFood = ""
lblTotal = ""
txtPrice.SetFocus
```


It should look exactly like this.



25. Close the code window using the .

Click on the Save icon. 

Code for Pet.frm Pet.vbp

26. Run the program by clicking on the **Start** icon  in the top toolbar. The cursor should be in the first textbox.
27. Use the following test data to see if your program gives the correct results.

Type 10 in the first textbox, then press **Tab**.

Type 20 in next textbox, then press **Tab**.

Type 30 in next textbox, then press **Tab**.

Type 40 in next textbox, then press **Tab**.


The cursor should now be on the **Calculate** button. Press **Enter**.

You should see 100 in the gray label that looks like a box. If you did not get 100, check the program for errors.



The screenshot shows a Windows-style application window titled "Pet by Miss Hangen (your name goes here)". The main title of the form is "Pet Expense". On the left side, there are five labels with corresponding textboxes: "Purchase Price \$" with value 10, "Vet Fee \$" with value 20, "Accessories \$" with value 30, "Food \$" with value 40, and "Total \$" with value 100. On the right side, there are three buttons: "Calculate", "Clear", and "Exit". The "Calculate" button is highlighted with a dashed border, indicating it is the current focus.

28. Press **Tab** to move to the **Clear** command button. Notice that the textboxes and the **lblTotal** label are now empty. The cursor should be in the first textbox.



The screenshot shows the same "Pet Expense" form, but now all the textboxes are empty. The "Total \$" label is also empty. The "Calculate" button is still highlighted. An arrow points from the top-left corner of the form to the first textbox, indicating the cursor's position.

Code for Pet.frm Pet.vbp

29. Test these values.

Pet Expense

Purchase Price \$	400	Calculate
Vet Fee \$	125.98	
Accessories \$	57.86	
Food \$	20.95	
		Clear
		Exit
Total \$	604.79	

30. Tab to the Exit command button and press Enter. The program is ended.

31. If you did not get the correct answer, look through the directions and through your program to find the errors; then correct them.

32. Click on the Save icon. 

Q5: What do these lines of code do?

```
txtPrice = ""  
txtFee = ""  
txtAccessories = ""  
txtFood = ""  
lblTotal = ""
```

A5: The textboxes and the answer label are assigned the `String` values with no characters in them. There is nothing between the quotation marks, so nothing will show in the boxes. `""` is called the *empty string*.

Q6: What does this line of code do?

```
txtPrice.SetFocus
```

A6: It moves the cursor to the `txtPrice` TextBox. What follows the dot after the TextBox name is the method that is being used. The `SetFocus` method puts the cursor in that TextBox.